A System for Secure Transfer of Data using Image Steganography

Asif Ansari[1] Ashish Bist[2] Sanskar Bandkar[3]
[1,2,3]Department of Computer Science & Engineering
[1,2,3]University of Mumbai, Mumbai, Maharashtra, India

Abstract— Algorithm and implementation of message hiding technique for secure transfer of data is described. The system which exists already doesn't integrate all the closely related technology for transferring a confidential data over an internet. In this paper, we proposed a system that integrates all the technology like encryption, transferring, storing and decryption, so that the job of user becomes increasingly easy and the task to maintain the system also becomes easy.

Key words: Image Processing, Data Hiding, Database, Steganography

I. INTRODUCTION

Steganography is the science of writing hidden messages in such a way that no one apart from the sender and intended recipient even realizes that there is a hidden message. Steganography is the practice of concealing a file, message, image, or video within another File, Message, Image, or Video. Steganography differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret. Steganography and cryptography are both ways to protect information from unwanted parties but neither technology alone is perfect and can be compromised. Once the presence of hidden information is revealed or even suspected, the purpose of steganography is partly defeated [2].

Image Steganography is the technique of hiding private or sensitive information within something that appears to be nothing be a usual image. Steganography involves hiding Text so it appears that to be a normal image or other file. To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image [1].

II. PROPOSED CONCEPT

Only encryption and hiding a data in an image is not enough, the system should be able to transfer that carrier to the receivers end with ease and without creating any doubt in the mind of the person who is snooping on the network to maximize the security, various levels of security is added to the system, so even if a person breaks into first layer, he will still have to break in two more layers which is almost impossible task. For example, before hiding the message into an image, the message is encrypted using an encryption algorithm like SHA1 etc., so even if a hacker is able to find that the image was a stegoimage he will find it very difficult to crack the encryption algorithm depending upon the robustness of the algorithm that we are implementing, it might take years for hackers to decrypt the message. And, at that point the message might become totally irrelevant to the context or to the situation. Apart from snooping, there is no way in which hacker can access the stegoimage even if he manages to gain access to the software application (which is only owned by sender and receiver). He won't be able to access the image as it will require the hacker to put username and password to login into the system. Now for the sake of discussion, we assume that by some means the hacker get access to credential of some user, and using it, it is logging into the system, he/she will still have to give two more input in order to access the image and get the decrypted form of original message. The following two inputs are key and another derivable integer number. This derivable number can be calculated only by using formula which is only known to legitimate members of the system and the complexity of the formula is so much that it will become almost impossible for an ignorant user to crack it.

III. METHOD

In order to hide a message into an image, we have to consider many factors as we are using pixels of an image to store data. The length of message that we can store in an image is limited to the number of available pixels in an image. Pixel count number will be large or small according to the size of an image, so if we are sending a long message than we have to use a large image, accordingly. Due to the fear of distorting the original image to a recognizable level we cannot use every pixel to store some data. That is why we have developed an algorithm (sort of pattern) that will decide which pixel should be used and there will be a fixed pattern according to which the selection will take place.

In this technique, we will be using an image as the carrier of the information. Each pixel of the image will carry some information. In this concept, the first pixel will carry the length of the message that is hidden. The red component of pixel is shifted two position to the right and the green pixel is shifted one position to the right. The information that can be stored in each component of the pixel is limited. This limit is not a pre-defined number, so it is not easy to go below and beyond it. But, in case if you cross that limit, the image will be distorted too much and it will be recognized by someone who is keeping track of the network.

We will alternatively select different color components of the pixels. Like we will select the red component of second pixel (first pixel is used for storing the length) for storing of information, for the next pixel we might select the green component of it. Similarly in the consecutive selection, we will use the blue component. We will go through all these three components in the round robin fashion and repeat this until we hide each and every byte of the secret message.

In this technique, we will not utilize each and every pixel of the image because it will change the composition of the image so much that it will be recognizable. So the whole point of steganography will become useless.

IV. RESULT

Thus we have successfully implemented an algorithm and embedded the data into a pixel of an image. As we can see in fig 4.1 we have used a very small white image and zoomed it to demonstrate the working of our algorithm. We

have also embedded large data in an image in fig 4.2, but the changes in the cover image and stego image cannot be spotted with the bare eye.

A. Non Coloured Image

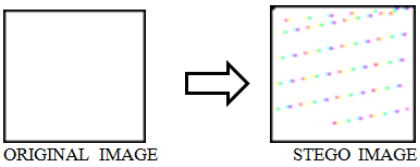

Fig. 4.1: Small non-coloured cover image and its corresponding stegoimage

B. Coloured Image

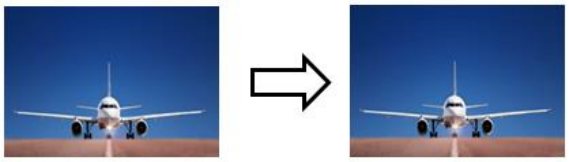

Fig. 4.2: Large coloured cover image and its corresponding stegoimage

V. Conclusion

In order to make the security of this application uncrackable we have added multiple levels of security, which includes (but not restricted to) a key that will be used for encryption and decryption purpose and it is shared by both sender and receiver. Apart from this, another derivable int/float number is used to make the security even more robust. This derivable number is calculated using a formula, known to both the parties, and this formula use the secret key as the only variable. In order to extract the hidden message from the stego-image, the recipient must punch in this calculated number as an additional credential.

Acknowledgment

The authors would like to thank and express deep gratitude towards Department of Computer Science and Engineering for providing us with excellent facilities that helped us to complete and present this paper.

References

[1] Champakamala B.S, Padmini.K, Radhika D. K Asst Professors, "Least Significant Bit algorithm for image steganography", Don Bosco Institute of Technology, Bangalore, India
[2] Wang, H & Wang, S, "Cyber warfare: Steganography vs. Steganalysis", Communications of the ACM, 47:10, October 2004

